MITRE-Bedford Division

LEVEL I

MTR-8201

Trusted Computer Systems-Glossary

(12) 24

George A. Huff

March 1981

MITRE

MITRE Technical Report

MTR-8201

Trusted Computer Systems-Glossary

George A. Huff

March 1981

CONTRACT SPONSOR OUSDRE (C^3I)
CONTRACT NO. F19628-81-C-0001
PROJECT NO 8420
DEPT. D75

THE MITRE CORPORATION
BEDFORD, MASSACHUSETTS

Department Approval: Edward H. Bensley

E. H. Bensley

MITRE Project Approval: Peter S. Tasker

P. S. Tasker

ABSTRACT

A Trusted Computer System is a computer system that employs sufficient hardware and software integrity measures to allow its use for the simultaneous processing of multiple levels of classified or sensitive information. This glossary was prepared for distribution at the Third Computer Security Initiative Seminar held at the National Bureau of Standards, November 18-20, 1980. Emphasis is on terms which relate to the formal specification and verification of such systems.

TRUSTED COMPUTER SYSTEMS GLOSSARY

This glossary was prepared for distribution at the Third Computer Security Initiative Seminar held at the National Bureau of Standards, November 18 20, 1980. Emphasis is on terms which relate to the formal specification and verification of trusted computer systems.

Many of these terms are used in other branches of computer science with differing or less specific meanings than those offered here. In the case of conflicting usage within the computer security community, preference is given to the most widely accepted meaning. Some terms whose definitions are self-evident have been omitted.

Access. The ability and the means necessary to store or retrieve data, to communicate with (that is, provide input to or receive output from), or otherwise make use of any resource in a computer system.

Access Control. A strategy for protecting objects from unauthorized access.

Access Control List. A list of subjects which are authorized to have access to some object. See object, subject.

Access Level. See security level.

Access Mode. A distinct operation recognized by the protection mechanisms as a possible operation on an object. Read, write and append are possible modes of access to a file, while execute is an additional mode of access to a program. See security mode.

Access Policy. See policy, DoD security policy.

Accreditation. The final acceptance of a system for operation in a specific environment. This is an administrative activity.

Accountability. The property that enables violations or attempted violations of system security to be traced to individuals who may then be held responsible.

Activity. A security model rule stating that once an object is made inactive, it cannot be accessed until it is made active again. See Bell-LaPadula security model.

Address Space. The virtual memory that can be addressed by a process. The maximum size of a process address space is usually a function of the underlying hardware.

Affirm. A formal methodology developed at the University of Southern California Information Sciences Institute (USC-ISI) for the specification and verification of abstract data types, incorporating algebraic specification techniques and hierarchical development.

Aggregation. A circumstance in which a totality of small pieces of information must be classified at a higher level than any single piece of information which comprises it.

Attention Character. In TCB design, a character that, when entered from a terminal, tells the TCB that the user wants a secure communications path from the terminal to some trusted code, in order to provide a secure service for the user, such as logging in or logging out.

Audit. An independent review and examination of system records and activities in order to test for adequacy of system controls, to ensure compliance with established policy and operational procedures, and to recommend any indicated changes in controls, policy and procedures.

Audit Trail. A chronological record of system activity which is sufficient to enable the reconstruction, review, and examination of the sequence of environments and activities surrounding or leading to each

event in the path of a transaction from its inception to output of final results.

Authenticate. To confirm the identity of a person (or other agent external to the protection system) making an access request.

Authentication. The act of identifying or verifying the eligibility of a station, originator or individual to access specific categories of information.
The process by which the Government determines concurrence with specifications.

Authorize. To grant a subject access to certain information.

Bell-LaPadula Security Model. An "access control" type of security model based on state-machine concepts; sometimes called the MITRE Model. In this model, the entities in a computer system are abstractly divided into sets of subjects (active entities such as processes) and objects (information containers). The notion of a secure state is defined, and an inductive proof of system security can be given: the initial system state is shown to be secure, and every state transition is shown to preserve this property.
A system state is defined to be "secure" if the only permitted accesses of subjects to objects are in accordance with specified security level restrictions. For example, a subject is permitted to read data at its own level or at a lower level (simple security condition), and to write data to its own level or to a higher level (*-property). State transitions preserve the "secure state" property in accordance with tranquility, erasure and activity principles (q.v.).
Among several versions of the Bell-LaPadula model is an integrity model, which is essentially the mathematical dual of the security model, incorporating a "simple integrity principle" and "integrity *-property." See integrity.

Capability. In a computer system, an unforgeable ticket that is accepted by the system as incontestable proof that the presenter has authorized access to the object named by the ticket. It is often interpreted by the operating system and the hardware as

an address for the object. Each capability also contains authorization information identifying the nature of the access mode (for example, read mode, write mode).

Category. The unit into which security information is partitioned, corresponding roughly to an interest group or topic area, for example, NATO, CRYPTO. Category information can exist at many levels, unclassified through top secret. A subject must be cleared to an adequate level and have access to the proper category or set of categories before access to classified data can be given. See security level.

Certification. The application of policy doctrine and technical evidence to a system to determine the prudence of its use in a particular secure application. This is a technical and political activity.

Certification refers to the "user" agreement, at the conclusion of the operational test and evaluation phase of a contract, that the acquired system satisfies its intended operational requirements.

Channel. An information transfer path within a system. "Direct" or "overt" channels are those paths that are designed for data transfer; "indirect" or "covert" channels are not explicitly intended for data transfer, but can pass information through the selective use of system resources, that is, through "leakage" or "signalling." Indirect channels are generally categorized as "storage" or "timing" channels. Timing channels are those that exploit the system clock or system performance characteristics to pass information and are difficult to identify in a nonprocedural system specification, while most storage channels can be identified by flow analysis techniques. Both types of indirect channels are exploitable only through interprocess cooperation.

Classification. The level of protection that must be afforded information. It is the information counterpart of clearance in DoD security policy. It applies to such system objects as buffers and files, as well as to "real-world" security-related documents.

Clearance. An authorization allowing a person access to

classified information. This is a "real-world" term used in connection with DoD policy, whose mathematical counterpart is security level (q.v.). A clearance typically consists of a level (unclassified through top secret) and a need-to-know category or categories. See security level.

Code Proof. See implementation verification.

Compartment. See category.

Computer Security. See security.

Compromise. See violation.

Confidentiality. The status accorded to private data and the degree of protection that must be provided for such data. Data confidentiality applies not only to data about individuals but to any proprietary or sensitive data that must be treated in confidence. See privacy.

Confinement. Allowing a process executing a borrowed program (in general, an arbitrary program) to have access to data, while ensuring that the data cannot be misused, altered, destroyed or released. See channel.

Confinement Channel. See channel.

Container. A repository of data in a system. See object.

Controlled Security Mode. A specific mode of system operation in government installations in which there are users who have legitimate access to the system but do not have either a security clearance or a need-to-know for all classified material contained in the system. Internal hardware and software must be provided and approved for maintaining separation of data and users with different classifications and clearances. No more than three adjacent security levels can be supported concurrently, and specific approval by the Designated Approving Authority is needed for this mode. See security mode.

Correctness. In a strict sense, the property of a system that is guaranteed as a result of formal

verification activities. Correctness is not an absolute property of a system, rather it implies the mutual consistency of a specification and its implementation. See verification.

Correctness Proof. A mathematical proof of consistency between a specification and its implementation. It may apply at the security model-to-formal specification level, at the formal specification-to-HOL code level, at the compiler level or at the hardware level. For example, if a system has a verified design and implementation, then its overall correctness rests with the correctness of the compiler and hardware.

Once a system is proved correct, it can be expected to perform as specified, but not necessarily as anticipated if the specifications are incomplete or inappropriate.

Covert Channel. See channel.

Data Security. Protection of data against accidental or deliberate modification, destruction or disclosure.

Dedicated Security Mode. In government installations, a mode of operation in which the computer system, its connected peripheral devices and remote terminals are exclusively used and controlled by specific users or groups of users who have a security clearance and need-to-know for all categories and types of classified material contained in the computing system. See security mode.

Denial of Service. The prevention of authorized access to computer resources, or the delaying of time-critical operations.

Design Verification. The use of verification techniques, usually computer-assisted, to demonstrate a mathematical correspondence between an abstract (security) model and a formal system specification. See verification.

Designated Approving Authority (DAA). The specified individual or agency which is responsible for the accreditation of a computer system. See accreditation.

Discretionary Access Controls. Access controls to an object that may be changed by the creator of the object. More generally, mechanisms that allow each subject, at its own discretion, to decide which of its own access rights are to be given to any other subject.

DoD Security Policy. The complete body of law, regulations and policy concerning the safeguarding of Defense sensitive information. DoD security policy includes all the espionage laws, the DoD regulations, and DoD authorized commercial classification for handling and access to information concerning national defense. The basic policy sets four levels and several categories of non-discretionary information control and requires that anyone accessing classified information have a "need-to-know" for the particular information in question. See non-discretionary security.

Domain (Hardware). A means by which hardware features can be restricted or nested. For example, the DEC PDP 11/45 or 11/70 has three execution domains: kernel, supervisor, and user. The kernel domain is the most privileged, and allows access to all hardware features including memory maps and I/O; the other domains do not allow these privileges. Another example is the Honeywell MULTICS architecture, which includes eight rings.

Domain (Software). An environment or context that defines the set of access rights that a subject has to objects of the system.

Downgrade. See regrade.

Emulator. A combination of hardware and software that permits programs written for one computer to be run on another computer.

In computer security terminology, the emulator is the portion of the system responsible for creating an operating system compatible environment out of the environment provided by the kernel. In KSOS, the emulator maps the kernel environment into the UNIX environment.

Encryption. The (usually) invertible coding of data through the use of a transformation key so that the

data can be safely transmitted or stored in a physically unprotected environment.

Erasure. A security model rule stating that objects must be purged before being activated or reassigned. This ensures that no information is retained within an object when it is reassigned to a subject at a different security level. See Bell-LaPadula security model.

Formal Development Methodology (FDM). See Ina Jo.

Flow Analysis. See security flow analysis.

Flow Control. A strategy for protecting the contents of information objects from being transferred to objects at improper security levels. It is more restrictive than access control. See access control, non-discretionary security.

Formal Specification. A specification of hardware or software in a computer-readable language, usually giving a precise mathematical description of the behavior of the system with the intention of providing support for formal verification. Formal specifications for a system can be written at any level of detail. See top level specification.

Granularity. The size of the smallest protectable unit of information. In a kernel-based system, this would be the size of the smallest protectable file or portion of virtual memory.

GYPSY. A combined formal program specification language and a verifiable high order language, developed at the University of Texas, and designed in conjunction with a complete verification system.

Hierarchical Development Methodology (HDM). A formal specification and verification methodology developed at SRI International. HDM is based on a nonprocedural, state-transition specification language, SPECIAL, and provides a security flow analysis tool, MLS, for verifying the multilevel security properties of a user-interface specification.

Human Interface Functions. TCB operations that require human intervention or judgment. Untrusted processes

would not be able to invoke them. See software interface functions, Trusted Computing Base.

Identification. The process that enables, generally by the use of unique machine-readable names, recognition of users or resources as identical to those previously described to the computer system.

Implementation Verification. The use of verification techniques, usually computer-assisted, to demonstrate a mathematical correspondence between a formal specification and its implementation in program code. See verification.

Ina Jo (also known as Formal Development Methodology). System Development Corporation's specification and verification methodology, based on a nonprocedural state-transition specification language, Ina Jo. The Ina Jo methodology incorporates user-supplied invariants to produce a formal demonstration that security properties are met.

Indirect Channel. See channel.

Integrity. The assurance, under all conditions, that a system will reflect the logical correctness and reliability of the operating system; the logical completeness of the hardware and software that implement the protection mechanisms; and the consistency of the data structures and accuracy of the stored data.

In a formal security model, integrity is interpreted more narrowly to mean protection against unauthorized modification or destruction of information.

Interface. The common boundary between independent systems or modules, where communication takes place.

Interprocess Communication (IPC). Communication between two different processes using system-supplied constructs; for example, shared files.

Isolation. The containment of users, data, and resources in an operating system in such a way that users may not access each other's data and resources, and may not manipulate the protection controls of the operating system. See confinement.

KSOS. Kernelized Secure Operating System. The project to strengthen the UNIX operating system with a security kernel to make it suitable for multilevel operations. See emulator.

KSOS-6. The KSOS implementation on Honeywell SCOMP hardware (a modified Level 6 Minicomputer).

KSOS-11. The KSOS implementation on the DEC PDP-11/45 and PDP-11/70 by Ford Aerospace and Communications Corporation.

KVM/370. Kernelized VM/370. The kernelized version of IBM's VM/370 for the S/370 series architecture, being built and verified by System Development Corporation.

Level. See security level.

Mandatory Security. See non-discretionary security.

MITRE Model. See Bell-LaPadula security model.

MITRE Secure UNIX. A prototype for a kernelized version of UNIX, developed at the MITRE Corporation. It was a forerunner of the KSOS systems. See UNIX, KSOS.

MLS. The multilevel security formula generator, a flow analysis tool developed at SRI for use with HDM. See Hierarchical Development Methodology, security flow analysis.

MULTICS. Multiplexed Information and Computing Service. A general-purpose timesharing system developed for a number of mainframes in the Honeywell Information Systems (HIS) line, among them the HIS 643 and 6180. MULTICS was enhanced to allow multilevel operation, and is presently running at the Air Force Data Services Center in controlled security mode (q.v.).

Multilevel Security. See DoD security policy, security level.

Multilevel Security Mode. A mode of system operation permitting data at various security levels to be concurrently stored and processed in a computer system where at least some users have neither the clearance nor the need-to-know for all classified

material contained on the system. Separation of personnel and material on the basis of security level is accomplished by the operating system and associated system software. See security mode.

Need-to-know. A job-related requirement for access to specific information. Need-to-know implies discretionary control of information--even though potential accessors may have the necessary clearance.

Non-discretionary Security. The aspect of DoD security policy which restricts access on the basis of security levels. A security level is composed of a level and a category set restriction. For read-access to an item of information, a user must have a clearance level greater than or equal to the classification of the information, and also have a category clearance which includes all of the access categories specified for the information. See discretionary access controls, security level.

Non-kernel Security-Related Software (NKSR). Security-relevant software which is executed in the environment provided by a security kernel, rather than as part of the kernel. Processes executing NKSR software may or may not require special privilege to override kernel-enforced security rules. See trusted process.

Nonprocedural Language. A formal high-level language for the specification of program modules. Such languages express relations which hold between "input" and "output" values of program variables, without constraining the particular algorithms which implement the change. See HDM, Ina Jo.

Object. In a formal security model, an identifiable resource, data container or related entity of the system; the counterpart of subject. Software-created entities such as files, programs and directories are objects, as well as hardware resources such as memory blocks, disk tracks, terminals, and tapes. See Bell-LaPadula security model, subject.

Password. A protected word or a string of characters that identifies or authenticates a user, a specific resource, or an access mode.

Penetration. The successful, repeatable, unauthorized extraction of recognizable information from a protected data file or data set.

Penetration Testing. The use of special programmer/analyst teams to attempt to penetrate a system for the purpose of identifying any security weaknesses.

Periods Processing. In computer installations, a mode of processing in which a specific security mode is temporarily established during a specific time interval for processing sensitive information. For example, the computer system could process secret information in the dedicated security mode during one period, and unclassified material in a second period. The computer system must be purged of all information before transitioning from one period to the next whenever there will be new users who do not have clearance and need-to-know for information processed during the previous period. See dedicated security mode.

Policy. Administrative decisions which determine how certain security-related concepts will be interpreted as system requirements. All such policy decisions must eventually be interpreted formally and implemented.

Privacy. The degree to which an individual or organization can decide whether, when, and to whom personal or organizational information is released.

Privileged Process. A process that is afforded (by the kernel) some privileges not afforded normal user processes. A typical privilege is the ability to override the security *-property. Privileged processes are trusted. See trusted process.

Process. The active system entity through which programs run. The entity in a computer system to which authorizations are granted; thus the unit of accountability in a computer system. A process consists of a unique address space containing its accessible program code and data, a program location for the currently executing instruction, and periodic access to the processor in order to continue.

Protection. Narrowly, the mechanisms used to control access of executing programs to stored data. See security.

PSOS. Provably Secure Operating System. A capability-based operating system structured as a hierarchy of nested abstract machines. PSOS was designed at SRI International and the system design utilizes SPECIAL and MLS. PSOS is now under development at Ford Aerospace and Communications Corporation. See capability, Hierarchical Development Methodology.

Recovery Procedures. The actions necessary to restore a system's computational capability and data files after a system failure or penetration.

Reference Monitor. A security control concept in which an abstract machine mediates accesses to objects by subjects. In principle, a reference monitor should be complete (in that it mediates every access), isolated from modification by system entities, and verifiable. A security kernel is an implementation of a reference monitor for a given hardware base.

Regrade. A determination that classified information requires a different degree of protection against unauthorized disclosure than currently provided, together with a change of classification designation that reflects such different degree of protection. An "upgrade" results in a higher classification; a "downgrade" results in a lower classification.

Ring. See domain.

Risk Analysis. The term applied to the systematic quantification of threats, loss exposures and countermeasure benefits.

Sanitize. To delete sensitive material from a file, or to modify it, so as to lower its security level. See regrade.

SCOMP. Secure Communications Processor. The name given to the Honeywell Level 6 Minicomputer modified to increase its protection capability. Four protection rings were added along with user-initiated input/output to direct-memory access devices. See KSOS-6.

Secure Path. See trusted path.

Security (Computer Security). Used in the most general sense, this denotes the totality of mechanisms and techniques that protect resources (including data and programs) from accidental or malicious modification, destruction, or disclosure. The term includes the physical security of the computer installation, administrative security, personnel security, and data security. Used more narrowly, it denotes protection of computer information from unauthorized disclosure.

Security Flow Analysis. A type of security analysis performed on a nonprocedural formal system specification which locates potential flows of information between system variables. By assigning security levels to system variables, many indirect information channels can be identified. Security flow analysis defines a security model similar to the access control model (Bell-LaPadula) but with a finer protection granularity.

Security Kernel. A localized mechanism, composed of hardware and software, that controls the access of users (and processes executing on their behalf) to repositories of information resident in or connected to the system. The correct operation of the kernel along with any associated trusted processes should be sufficient to guarantee enforcement of the constraints on access. TCBs have been implemented using security kernels along with trusted processes. See Trusted Computing Base.

Security Level. In the context of formal security modeling, the fundamental security attribute of subjects and objects. Security levels combine a level (e.g., Unclassified, Confidential, Secret, Top Secret) and a set of need-to-know categories. A derived partial ordering of security levels is defined using a combination of the "less than" order on levels and the "subset" relation on category set. Thus (Secret, {NATO}) is less than both (Top Secret, {NATO}) and (Secret, {NATO, CRYPTO}). The security levels (Confidential, {NATO}) and (Top Secret, {CRYPTO}) are incomparable (that is, a user at one of these levels cannot access information at the other level). This derived partial ordering on security levels is the

basis on which all subject-to-object access is determined.

Security Mode. A Department of Defense term for "Authorized variations in the security environments and methods of operating ADP systems that handle classified data." DoD ADP security policy (DoD Directive 5200.28) defines four modes: dedicated, system high, controlled and multilevel.

Security Model. See Bell-LaPadula security model, correctness, security flow analysis, verification.

Security Policy. See DoD security policy, nondiscretionary security.

Security Violation. See violation.

Security *-property (pronounced "star" property). A security model rule allowing a subject write-access to an object only if the security level of the object is the same or higher than the security level of the subject. See Bell-LaPadula security model.

Simple Security Condition. A security model rule allowing a subject read-access to an object only if the security level of the object is the same or less than the security level of the subject. See Bell-LaPadula security model.

Software Interface Functions. TCB operations that can be invoked by software. See human interface functions, Trusted Computing Base.

SPECIAL. See Hierarchical Development Methodology.

Spoofing. The deliberate inducement of a user or a resource to take an incorrect action.

Specification. Generally, a description of the input, output and essential functions to be performed by a system or by a component of a system. The specification is produced by the organization that is to develop the system; hence at the top level it can be thought of as the contractor's interpretation of the requirements. See formal specification.

Storage Channel. See channel.

Subject. An active user of a computer system together with any other entity acting on behalf of a user or on behalf of the system; for example, processes, jobs and procedures may all be considered subjects. Certain subjects may also be considered to be objects of the system. See Bell-LaPadula security model, object.

System High. The highest security level supported by a system at a particular time or in a particular environment.

System High Security Mode. The mode of operation in which the computer system and all of its connected peripheral devices and remote terminals are protected in accordance with the requirements for the highest security level of material contained in the system at that time. All personnel having computer system access must have the security clearance and need-to-know for all material then contained in the system.

System Low. The lowest security level supported by a system at a particular time or in a particular environment.

System Utility. Any software, not part of the verified operating system, used to perform various functions that are not directly part of the applications. Examples are compilers, debuggers, editors, and loaders.

Timing Channel. See channel.

Top Level Specification (TLS). In a verification context, a mathematical specification of system behavior at the most abstract level, typically a functional specification that omits all implementation detail. The formal top level specification of a security kernel precisely defines the behavior of the security kernel observable outside the kernel domain (at the kernel interface). See formal specification.

Tranquility. A security model rule stating that the security level of an active object does not change. See Bell-LaPadula security model.

Trap Door. An entry point in a computer system that can be selectively accessed to allow unauthorized access to the system.

Trojan Horse. A borrowed program that performs actions unrelated to the caller's intent, subverting the security of the caller's data. It may disclose sensitive data either by hiding it in a file or other form of storage where it can be accessed later, or by communicating the information via a covert channel. The name Trojan Horse was given to this kind of problem because it involves a foreign or gift program which has unexpected or malicious side effects. Trojan Horses may be planted on a temporary or permanent basis. See channel, confinement.

Trusted Computing Base (TCB). The totality of protection mechanisms for an operating system. It provides both a basic protection environment plus additional user services required for a trustworthy turnkey system. TCBs have been implemented as security kernels and trusted processes.

Trusted Path. A direct and reliable connection between a user at a terminal and a trusted process. A trusted path may be activated through an attention character. Also called secure path. See attention character.

Trusted Process. A process which can affect system security. It is sometimes but not always endowed with privileges to override kernel-enforced rules. The protection capabilities or characteristics of a trusted process must be reliably demonstrated to comply with stated requirements, through formal verification when possible. Trusted processes are sometimes used to execute NKSR software.

UCLA Data Secure UNIX. A prototype for a kernelized version of UNIX, developed at UCLA. It was a forerunner of KSOS. See UNIX, KSOS.

Unauthorized Disclosure. See violation.

UNIX. A general-purpose time-sharing operating system designed and built by Bell Laboratories and intended originally for use with the DEC PDP-11 series

computer. Secure system developments have been based on UNIX (KSOS-6 and KSOS-11), and both UCLA and MITRE have designed secure UNIX prototypes.

Untrusted Process. A process whose incorrect or malicious execution cannot affect system security. Verification is usually not applied to untrusted processes.

Upgrade. See regrade.

Validation. The collection of evaluation, integration and test activities carried out at the system level to ensure that the system being developed satisfies the requirements of the system specification.

Verification. Informally, a clear and convincing demonstration that software is correct with respect to well-defined criteria, such as a security model. In a formal context, verification refers to the mathematical demonstration of consistency between a formal specification and a security model (design verification) or between the formal specification and its program implementation (implementation verification). The phrase "formally verified" is now beginning to imply that computer-assisted techniques have been employed in the verification effort. See correctness.

Violation. Some form of security breach. "Compromise" carries the connotation that security-relevant data has "possibly" been exposed to unauthorized persons (or processes), while "unauthorized disclosure" implies that the data has actually been read.

Virtual Space. See address space.

*-Property. See security *-property.

BIBLIOGRAPHY

R. P. Abbott et al, Security Analysis and Enhancements of Computer Operating Systems, Institute of Computer Sciences and Technology, National Bureau of Standards, Washington DC, April, 1976.

S. A. Gloss-Soler, The DACS Glossary, A Bibliography of Software Engineering Terms, Data and Analysis Center for Software, RADC/ISISI, Griffiss AFB, NY, October, 1979.

L. J. Hoffman, Modern Methods for Computer Security and Privacy, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1977.

R. W. Jensen and C. C. Tonies, Software Engineering, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1979.

T. A. Linden, "Operating System Structures to Support Security and Reliable Software," Computing Surveys, Vol. 8, No. 4, December, 1976.

J. H. Saltzer and M. D. Schroeder, "The Protection of Information in Computer Systems," Proceedings of the IEEE, September 1975.

Glossary for Computer Systems Security, Federal Information Processing Standards Publication 39, National Bureau of Standards, February, 1976.

"Security Requirements for Automatic Data Processing (ADP) Systems," Department of Defense Directive 5200.28, December, 1972; Amended, April, 1978.

DISTRIBUTION LIST

D-40

J. Mitchell
C. M. Sheehan

D-44

M. Ferdman

D-60

S. B. Lipner

D-70

W. S. Attridge
E. L. Lafferty
W. S. Melahn

D-71

S. R. Ames
J. B. Glore
M. Hazle

D-72

M. J. Corasick
J. C. C. White

D-73

T. L. Connors

D-75

D. L. Baldauf
D. J. Baughman
D. Benaroya
E. H. Bensley
E. L. Burke
M. H. Cheheyl
A-M. G. Discepolo

D-75 (Continued)

D. L. Drake
K. B. Gasser
M. Gasser
R. A. J. Gildea
R. D. Graubart
G. A. Huff (15)
J. G. Keeton-Williams
K. E. Kirkpatrick
S. M. Kramer
C. W. McClure
J. K. Millen
G. H. Nibaldi
J. J. O'Connor
R. S. Popp
S. A. Rajunas
M. J. Reece
D. P. Sidhu
D. J. Solomon
S. H. Stahl
P. S. Tasker (15)
E. T. Trotter
E. E. Wiatrowski
W. F. Wilson
J. P. L. Woodward